

This listing of claims will replace all prior versions, and listings, of claims in the application.

**Listing of Claims:**

Claims 1-22 (canceled)

23. (currently amended) ~~A method according to claim 19, A method of compiling programming language code with a compiler, comprising:~~  
~~initially parsing the code;~~  
~~during said initial parsing, encountering an attribute block and allowing an interface definition that follows the attribute block to include embedded information; and~~  
~~parsing the interface definition wherein when the compiler encounters a soap\_handler attribute, the compiler recognizes soap\_handler as an attribute that is supported by an attribute provider and calls into the attribute provider.~~

24. (original) A method according to claim 23, wherein the attribute provider marks the class on which the soap\_handler attribute appears as handling (Simple Object Access Protocol) SOAP messages and then the attribute provider returns control to the compiler.

25. (original) A method according to claim 24, wherein after said return of control, the compiler begins parsing the body of the class on which the soap\_handler attribute appears and parses each function in the class.

26. (original) A method according to 25, wherein for each soap\_header attribute on the function, the compiler performs soap\_header attribute processing.

27. (previously presented) A method according to claim 26, wherein said soap\_header attribute processing includes recognizing a soap\_header attribute as an attribute that is supported by the attribute provider and calling into the attribute provider whereby the attribute provider (i) gathers information to create the “Header” part of a SOAP message, (ii) queries the compiler for information about the parameters to the soap\_header attribute and (iii) queries the compiler for information about the data type for the header.

28. (original) A method according to claim 26, wherein said soap\_header attribute processing includes generating the information for creating a portion of the (Simple Object Access Protocol) SOAP message header and returning control from the attribute provider to the compiler.
29. (previously presented) A method according to 25, wherein if the function has a soap\_method attribute, the compiler performs soap\_method attribute processing.
30. (original) A method according to claim 29, wherein said soap\_method attribute processing includes recognizing a soap\_method attribute as an attribute that is supported by the attribute provider and calls into the attribute provider whereby the attribute provider (i) gathers information to create a (Simple Object Access Protocol) SOAP message for the function and (ii) queries the compiler for information about the parameters to the function on which soap\_method attribute appears.
31. (original) A method according to claim 30, wherein for each parameter on the function, the attribute provider (i) uses the interface definition information to determine which function parameters are to be received as part of the incoming SOAP message, and which parameters are to be sent back as part of the SOAP message response and (ii) queries the compiler about the data type of the parameter.
32. (original) A method according to claim 29, wherein said soap\_method attribute processing includes generating the data to create the SOAP message portion for a parameter and returning control from the attribute provider to the compiler.

Claims 33-42 (canceled)